

# CSCI 305 Homework 1

---

**Due Date: February 9, 2018 @ Beginning of Class**

Name: Chenwei Cao

## History

**Instruction:** Select the single best answer (unless directed otherwise by the question) which answers the question.

1. What is the name of the paradigm that is based on formal logic and expresses facts and rules about some problem domain? (1 pt)
  - a. functional
  - b. concurrent
  - ☒ c. logic
  - d. non-deterministic
  - e. metaprogramming
  - f. object-oriented
  - g. declarative
  - h. imperative

2. What is the name of the paradigm that treats computation as the evaluation of mathematical functions and avoids changing-state and mutable data? (1 pt)

- a. event-driven
- b. concurrent
- c. scripting
- d. object-oriented
- e. logic
- f. imperative
- g. declarative
- ☒ h. functional
- i. metaprogramming
- j. non-deterministic

3. What language was the first to introduce some object-oriented concepts (but is not a fully object-oriented language)? (1 pt)

- a. COBOL
- ☒ b. LISP
- c. Fortran
- d. Simula
- e. Smalltalk
- f. PL/I
- g. Ada
- h. Algol
- i. BASIC
- j. C

4. What language is considered to be the first fully object-oriented language? (1 pt)

- a. Smalltalk
- b. C
- c. BASIC
- ☒ d. Simula
- e. Algol
- f. PL/I
- g. Fortran
- h. Ada
- i. COBOL
- j. LISP

5. Research "typeless language". Select **ALL** from the below that are **DISadvantages** of a typeless language (2 pts)

- a. It is often useful for very low-level languages used for systems programming
- b. it provides great flexibility for the programmer.
- c. Programmers can make changes to existing code in less time.
- ☒ d. Type checking is impossible; it is the programmer's responsibility to ensure that expressions and assignments are correct.
- ☒ e. Reading the code of another person may be more difficult.
- f. Such a language is easier to learn as a first programming language.
- ☒ g. It leads to an increase in the possibility of run-time errors.
- ☒ h. It results in poor reliability in type conversion and checking.

## Grammar and Syntax

1. Construct a BNF grammar for each of the following languages. For example, a correct answer for "the set of all strings consisting of zero or more concatenated copies of the string `ab`" would be:

```
<S> ::= ab <S> | <empty>
```

- a. The set of all strings consisting of the keyword `begin`, followed by one or more statements with a semicolon after each one, followed by the keyword `end`. Use the non-terminal `<statement>` for statements, and do not give productions for it. (2 pts)

$$\langle S \rangle ::= \text{begin } \langle \text{statement} \rangle \text{ end}$$
$$\langle \text{statement} \rangle ::= \langle \text{statement} \rangle ; \mid \langle \text{statement} \rangle \langle \text{statement} \rangle ;$$

- b. The set of all strings consisting of an open bracket (the symbol `[`) followed by a list of one or more digits separated by commas, followed by a closing bracket (the symbol `]`). (2 pts)

$$\langle S \rangle ::= [ \langle \text{number} \rangle ]$$
$$\langle \text{number} \rangle ::= \langle \text{digit} \rangle , \langle \text{number} \rangle \mid \langle \text{digit} \rangle$$
$$\langle \text{digit} \rangle ::= 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9$$

2. Construct an EBNF grammar for each of the languages in Grammars Question 1. Use the EBNF extensions wherever possible to simplify the grammars. In particular, you should eliminate explicit recursion from the grammars wherever possible. Don't forget to put single quotes around any EBNF metasympol when using it as a token.

- a. (2 pts)

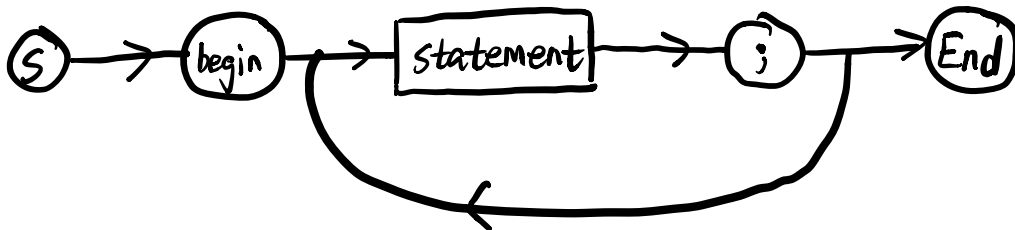
$$\langle S \rangle ::= \text{begin } \langle \text{statement} \rangle ; \{ \langle \text{statement} \rangle ; \} \text{ end}$$

b. (2 pts)

$\langle S \rangle ::= [ \langle \text{digit} \rangle \{, \langle \text{digit} \rangle \} ]$   
 $\langle \text{digit} \rangle ::= 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9$

3. Construct a syntax diagram for each of the languages in Grammars Question 1. Use branching and loops in your syntax diagrams to make them as clear as possible.

a. (3 pts)



b. (3 pts)

